TECHNISCHE
UNIVERSITÄT
DRESDEN

Technische Universitat Dresden
Institute for Theoretical Computer Science
Chair for Automata Theory

LTCS–Report

Similarity Measures for Computing Relaxed Instances w.r.t. General $\mathcal{EL}$-TBoxes

Andreas Ecke   Anni-Yasmin Turhan

LTCS-Report 13-12

Postal Address:
Lehrstuhl fur Automatentheorie
Institut fur Theoretische Informatik
TU Dresden
01062 Dresden

http://lat.inf.tu-dresden.de

Visiting Address:
Nothnitzer Str. 46
Dresden

Similarity Measures for Computing Relaxed Instances w.r.t. General $\mathcal{EL}$-TBoxes

Andreas Ecke* Anni-Yasmin Turhan

December 2013

Abstract

The notion of concept similarity is central to several ontology tasks and can be employed to realize relaxed versions of classical reasoning services. In this paper we investigate the reasoning service of answering instance queries in a relaxed fashion, where the query concept is relaxed by means of a concept similarity measure (CSM). To this end we investigate CSMs that assess the similarity of $\mathcal{EL}$-concepts defined w.r.t. a general $\mathcal{EL}$-TBox. We derive such a family of CSMs from a family of similarity measures for finite interpretations and show in both cases that the resulting measures enjoy a collection of formal properties. These properties allow us to devise an algorithm for computing relaxed instances w.r.t. general $\mathcal{EL}$-TBoxes, where users can specify the 'appropriate' notion of similarity by instanciating our CSM appropriately.

1 Introduction

Description Logics (DL) are a family of knowledge representation formalisms used to describe terminologies in various application areas in a way that can be used for reasoning. For this, each DL provides a set of constructors that can be used to create concept descriptions from sets of concepts and role names. The terminological knowledge is captured by describing relations between these concept descriptions. These axioms are collected in a so-called TBox. Additionally, DLs allow to model specific individuals as instances of concepts from the TBox, and

*Supported by DFG Graduiertenkolleg 1763 (QuantLA).

specific relationships between different individuals in the so-called ABox. TBox and ABox together form a knowledge base (KB). Besides providing the user with a formal language to model their application domains, DL systems also provide a variety of inference services, e.g. computing sub-concept relationships between the concepts described by the terminology or checking whether individuals given in the ontology are instances of a given concept.

However, in many cases, these standard inferences are too restricted. Assume for example a service platform, where individual services running on different platforms are described by means of a DL knowledge base. Clients want to select different services based on their preferences and requirements, for example by specifying a query concept that is matched against all services and returns exactly those, that meet all requirements (the instances of the query concept). However, if no service matches the query concept, it can be still be preferable to return similar services. These would only fulfill most of the requirements, but cannot guarantee e.g. the specified QoS, instead of returning no services at all. This kind of reasoning service is what we call relaxed instance query answering.

Ideally, it needs to be possible to specify which features are deemed more important, and which features might be relaxed. Additionally, one needs to be able to specify, how much relaxation is allowed. Both of this can be solved by using a concept similarity measure (CSM). A *concept similarity measure* assigns to a pair of concepts a value between 0 and 1, where a higher value indicates stronger similarity. Now, to guide the relaxation by the use of CSMs, also a threshold is needed that sets the minimal required degree of similarity. Matching individuals must be similar to the query concept w.r.t. this degree. In this setting, especially parameterizable similarity measures are useful, as they allow users to adjust their measure according to preferences, requirements and application task. For instance, by assigning different weights to different features of a concept or by specifying a primitive similarity between different names appearing in the concepts.

CSMs are an interesting ontology services in their own right. They play an important role for ontology alignment, where for two given ontologies the task is to find the corresponding concepts across the ontologies. Furthermore CSMs are used in the bio-medical field. In context of the Gene Ontology [6] CSMs are employed to find genes that are similar and thus might realize a similar functionality [11, 13].

However, previously defined concept similarity measures often lack formal semantics (see [9] for a study on this). Most CSMs for DLs can be divided in two groups: The first group, *structural measures*, compute the similarity value by recursively following the syntax trees of concepts. *Semantic similarity measures*, on the other hand, use the interpretation of concepts, e.g. the set of all objects that the concepts have as instance, to compare them. This however means, that primitive concepts that cannot be distinguished by the modeled objects always have the similarity 1. However, just because for a certain service provider, the services that provide a certain functionality are exactly those that run on Linux servers, doesn't mean that we want to treat those concepts as equivalent.

For structural CSMs properties such as equivalence invariance, i.e., treating equivalent concepts exactly the same, are—to the best of our knowledge—only investigated for concepts or concepts defined w.r.t. unfoldable TBoxes (i.e., terminologies). In this case the concepts using defined concepts can be expanded w.r.t. the unfoldable TBox. Then the syntax tree of the expanded concepts can be traversed to obtain the similarity without considering the TBox any further. This unfolding-based approach is not possible as soon as cyclic or general TBoxes are considered.

This paper pursues two main goals. The first is to introduce a new parameterizable, structural CSM that assesses the similarity of concepts defined w.r.t. general $\mathcal{EL}$-TBoxes, while taking the whole information from the TBox into account. There exists prior work on CSMs that work in regard of general TBoxes, but these are merely using the concept hierarchy (e.g. [1, 4]) and not the complete information from the TBox. For our CSM w.r.t. general $\mathcal{EL}$-TBoxes, we use the canonical models of the TBox and the concepts that should be compared. Then, one can use similarity measures on the elements of those canonical models to derive a similarity value for the concepts. We define $\sim_i$, a family of *interpretation similarity measures* that are parameterizable in several ways. We transfer the useful properties given in [9] for CSMs to interpretation similarity measures and show that the measure $\sim_i$ exhibits many of these properties. Based on $\sim_i$ we define a parameterizable CSM such that these properties are preserved. The second goal of this paper is to show how the problem of relaxed instance query answering can be solved w.r.t. general $\mathcal{EL}$-TBoxes. To this end we employ our newly introduced CSM derived from $\sim_i$.

The paper is structured as follows: In the next section we introduce the basic notions required; this includes the description logic $\mathcal{EL}$ and various inferences; the notion of simulations and canonical models in $\mathcal{EL}$ and their connection to subsumption; and finally concept similarity measures and their properties. Section 3 defines interpretation similarity measures (ISM) and transfers the properties given for CSMs to this setting. We introduce our ISM $\sim_i$ and show that $\sim_i$ is a well-defined measure that enjoys many of these properties. Based on this ISM, Section 4 will introduce the concept similarity measure $\sim_c$ that works on general $\mathcal{EL}$-TBoxes. This measure is then used to show how relaxed instance queries can be answered w.r.t. general $\mathcal{EL}$-TBoxes. We devise an iterative algorithm, for which we show

Constructor	Syntax	Semantics
top concept	$\top$	$\Delta^{\mathcal{I}}$
concept name	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{d \in \Delta^{\mathcal{I}} \mid \exists e.(d,e) \in r^{\mathcal{I}} \wedge e \in C^{\mathcal{I}}\}$
general concept inclusion	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
concept assertion	$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
role assertion	$r(a,b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$

Table 1: Concept constructors, TBox axioms and ABox assertions for $\mathcal{EL}$.

soundness and completeness and discuss its complexity. We end the report with some conclusions and pointer to future work.

2 Preliminaries

In this section we introduce the basic notions of Description Logics and some of the inference services and properties that we need throughout the paper. We start by defining the syntax and semantics of the lightweight DL $\mathcal{EL}$.

2.1 The Description Logic $\mathcal{EL}$

$\mathcal{EL}$-concept descriptions are constructed from two countable sets: The set N_C of *concept names* and the set N_R of *role names*. Given these, the constructors from the upper part of Table 1 can be applied to build complex concept descriptions. For example, the concept

$$\mathsf{Human} \sqcap \exists \mathsf{gender}.\mathsf{Male} \sqcap \exists \mathsf{hasChild}.(\mathsf{Human} \sqcap \exists \mathsf{gender}.\mathsf{Female})$$

describes men who have a daughter. With $\mathfrak{C}(\mathcal{EL})$ we denote the set of all $\mathcal{EL}$-concept descriptions.

Using these concept descriptions, one can specify the domain knowledge as a *TBox*, which is a set of *general concept inclusions* (GCIs, see again Table 1). Basically, a GCI $C \sqsubseteq D$ says that anything that belongs to concept C must also belong to concept D. We use the notion $C \equiv D$ as an abbreviation for the two GCIs $C \sqsubseteq D$ and $D \sqsubseteq C$, to express that the concepts C and D are equivalent. For example, the following TBox expresses that a man is defined as a human with male gender and that a grandfather always has a child (but not every person that has a child is a grandfather):

$$\begin{aligned} \mathcal{T} = \{&\mathsf{Man} \equiv \mathsf{Human} \sqcap \exists \mathsf{gender}.\mathsf{Male}, \\ &\mathsf{Grandfather} \sqsubseteq \exists \mathsf{hasChild}.\top\} \end{aligned}$$

Semantics of $\mathcal{EL}$-concept descriptions are introduced by means of *interpretations* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, consisting of a non-empty set $\Delta^{\mathcal{I}}$ of elements, called the *domain*, and an *interpretation function* $\cdot^{\mathcal{I}}$ which assigns subsets of $\Delta^{\mathcal{I}}$ to concept names and binary relations over $\Delta^{\mathcal{I}}$ to role names. We denote the set of all interpretations as $\mathfrak{I}$. This interpretation function of an interpretation $\mathcal{I}$ is extended to the set of concept descriptions as shown in Table 1. We say that an interpretation $\mathcal{I}$ satisfies the GCI $C \sqsubseteq D$, denoted $\mathcal{I} \models C \sqsubseteq D$, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, and it is a *model of the TBox* $\mathcal{T}$, denoted $\mathcal{I} \models \mathcal{T}$, if it satisfies all GCIs occurring in $\mathcal{T}$.

Description Logics also allow to describe individual objects. For this, another set N_I of *individual names* is needed. Using these, one can express that an individual a is an instance of a concept description C using the *concept assertion* $C(a)$, and that two individuals a and b are related via a role r using the *role assertion* $r(a, b)$. Interpretations $\mathcal{I}$ then additionally assign an element of the domain $\Delta^{\mathcal{I}}$ to each individual name, and assertions are interpreted as expected (see the lower part of Table 1). A set of concept and role assertions is called an *ABox*. Similarly to TBoxes, we say that an interpretation $\mathcal{I}$ is a model of an ABox $\mathcal{A}$, denoted $\mathcal{I} \models \mathcal{A}$, if it satisfies all assertions occurring in $\mathcal{A}$.

Taken together, an ABox $\mathcal{A}$ and a TBox $\mathcal{T}$ result in a *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$. An interpretation $\mathcal{I}$ is a model of a knowledge base $\mathcal{K}$, $\mathcal{I} \models \mathcal{K}$, if $\mathcal{I}$ is a model of both $\mathcal{T}$ and $\mathcal{A}$. With $\mathrm{Sig}(\mathcal{K})$ we denote the signature of a knowledge base $\mathcal{K}$, i.e., the set of all concept, role, and individual names occurring in $\mathcal{K}$. We write $\mathrm{Sig}_C(\mathcal{K})$, $\mathrm{Sig}_R(\mathcal{K})$, and $\mathrm{Sig}_I(\mathcal{K})$ instead of $\mathrm{Sig}(\mathcal{K}) \cap N_C$, $\mathrm{Sig}(\mathcal{K}) \cap N_R$, and $\mathrm{Sig}(\mathcal{K}) \cap N_I$, to refer to only one type of name. Similarly, we denote the signature of a TBox $\mathcal{T}$, an ABox $\mathcal{A}$ and a concept description C with $\mathrm{Sig}(\mathcal{T})$, $\mathrm{Sig}(\mathcal{A})$, and $\mathrm{Sig}(C)$, respectively.

DL systems offer a variety of reasoning services. Core inferences, that most systems provide, are subsumption and instance checking. *Subsumption* tests, given a TBox $\mathcal{T}$ and two concept descriptions C and D, whether C subsumes D w.r.t. $\mathcal{T}$ (denoted as $C \sqsubseteq_{\mathcal{T}} D$), i.e., whether $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$. Similarly, for a given knowledge base $\mathcal{K}$, an individual a and a concept description C, *instance checking* tests whether a is an instance of C w.r.t. $\mathcal{K}$ (denoted $\mathcal{K} \models C(a)$), i.e., whether $a^{\mathcal{I}} \in C^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{K}$. Given a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ and a concept description

C, *instance retrieval* returns all individuals from $\mathcal{A}$ that are instances of C. This last instance is sometimes also called *instance queries* and is the one for which we want to provide a relaxed version in this paper. However, there also exist useful non-standard inferences, like the most specific concept.

Definition 1 (most specific concept)**.** Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{EL}$-knowledge base, C be an $\mathcal{EL}$-concept description, and a be an individual occurring in $\mathcal{A}$. The $\mathcal{EL}$-concept description M is the *most specific concept* of a w.r.t. the KB $\mathcal{K}$ (denoted $\text{msc}_{\mathcal{K}}(a)$), iff:

1. $\mathcal{K} \models M(a)$, and
2. for all $\mathcal{EL}$-concept descriptions E with $\mathcal{K} \models E(a)$ we have $M \sqsubseteq_{\mathcal{T}} E$.

For $\mathcal{EL}$, the msc may not always exist due to cycles in the TBox or between individuals in the ABox, but if it does exist, it is unique up to equivalence, see [2, 12].

Consider for example the knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ with:

$$\begin{aligned} \mathcal{T} &= \{\ \mathsf{C} \sqsubseteq \mathsf{A} \sqcap \exists \mathsf{r}.\mathsf{B}, \quad \mathsf{D} \sqsubseteq \mathsf{C} \sqcap \exists \mathsf{r}.\mathsf{D}\ \} \\ \mathcal{A} &= \{\ \mathsf{D}(\mathsf{a}), \quad \mathsf{C}(\mathsf{b}), \quad \mathsf{r}(\mathsf{a}, \mathsf{b}), \\ &\qquad \mathsf{B}(\mathsf{c}), \quad \mathsf{r}(\mathsf{c}, \mathsf{c})\ \}. \end{aligned}$$

Then the msc of a w.r.t. $\mathcal{K}$ is $\mathsf{D} \sqcap \exists \mathsf{r}.\mathsf{C}$, while the msc of c would be an infinite concept description $\mathsf{B} \sqcap \exists \mathsf{r}.(\mathsf{B} \sqcap \exists \mathsf{r}.(\mathsf{B} \sqcap \exists \mathsf{r}. \ldots))$ and therefore does not exist in $\mathcal{EL}$.

2.2 Simulations and Canonical Models

Another useful notion when dealing with interpretations in DLs are simulations, which relate elements of different interpretations and characterize inferences like subsumption and generalizations. In the following, we describe those simulations used to describe inferences in $\mathcal{EL}$.

Definition 2 (simulation)**.** Let $\mathcal{I}$ and $\mathcal{J}$ be interpretations. A relation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$ is a *simulation* between $\mathcal{I}$ and $\mathcal{J}$, if the following conditions hold:

1. for all $(d, e) \in S$ and $A \in N_C$, if $d \in A^{\mathcal{I}}$, then $e \in A^{\mathcal{J}}$; and
2. for all $(d, e) \in S$, $r \in N_R$ and $(d, d') \in r^{\mathcal{I}}$, there is an $(e, e') \in r^{\mathcal{J}}$ with $(d', e') \in S$.

A *pointed interpretation* $p = (\mathcal{I}, d)$ consists of an interpretation $\mathcal{I}$ and a designated element $d \in \Delta^{\mathcal{I}}$. With $\mathfrak{P}$ we denote the set of all pointed interpretations, i.e. $\mathfrak{P} = \{(\mathcal{I}, d) \mid \mathcal{I} \in \mathfrak{I}, d \in \Delta^{\mathcal{I}}\}$. Given a pointed interpretation $p = (\mathcal{I}, d)$, we denote with $\mathfrak{C}(p) = \{C \in \mathfrak{C}(\mathcal{EL}) \mid d \in C^{\mathcal{I}}\}$ the set of all $\mathcal{EL}$-concept descriptions that have d as an instance in $\mathcal{I}$.

For two pointed interpretations $p = (\mathcal{I}, d)$ and $q = (\mathcal{J}, e)$, we say that p *simulates* q (denoted $p \lesssim q$), if there exists a simulation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$ between the interpretations $\mathcal{I}$ and $\mathcal{J}$ with $(d, e) \in S$. If p and q simulate each other, i.e., $p \lesssim q$ and $q \lesssim p$, we write $p \simeq q$ and say that p and q are *equisimilar*.

In [10] the strong connection between simulations, pointed interpretations and their concept sets were shown.

Theorem 3 ([10])**.** *Let p and q be two pointed interpretations. Then:*

1. *$p \lesssim q$ iff $\mathfrak{C}(p) \subseteq \mathfrak{C}(q)$, and*
2. *$p \simeq q$ iff $\mathfrak{C}(p) = \mathfrak{C}(q)$.*

For an $\mathcal{EL}$ knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, one can create a canonical model for a concept description C or the ABox $\mathcal{A}$ with respect to the TBox $\mathcal{T}$, called $\mathcal{I}_{C,\mathcal{T}}$ and $\mathcal{I}_{\mathcal{K}}$. The canonical model $\mathcal{I}_{C,\mathcal{T}}$ is always a model of $\mathcal{T}$ and contains an element $d_C \in \Delta^{\mathcal{I}_{C,\mathcal{T}}}$ which is an instance of C. Similarly, $\mathcal{I}_{\mathcal{K}}$ is always a model of both the TBox $\mathcal{T}$ and the ABox $\mathcal{A}$, and, necessarily, contains an element $d_a \in \Delta^{\mathcal{I}_{\mathcal{K}}}$ for each individual a in $\mathcal{A}$ such that $d_a = a^{\mathcal{I}_{\mathcal{K}}}$.

Definition 4 (canonical models)**.** Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{EL}$-KB consisting of a TBox $\mathcal{T}$ and ABox $\mathcal{A}$, and C be an $\mathcal{EL}$-concept description. With $\mathrm{sub}(C)$ ($\mathrm{sub}(\mathcal{T})$, $\mathrm{sub}(\mathcal{A})$) we denote the set of all subconcepts of the concept C (all subconcepts of concepts occurring in $\mathcal{T}$ or $\mathcal{A}$, respectively).

The *canonical model $\mathcal{I}_{C,\mathcal{T}}$ of C* w.r.t. the TBox $\mathcal{T}$ consists of the domain $\Delta^{\mathcal{I}_{C,\mathcal{T}}}$ and the interpretation function $\cdot^{\mathcal{I}_{C,\mathcal{T}}}$ defined as follows:

- $\Delta^{\mathcal{I}_{C,\mathcal{T}}} = \{d_C\} \cup \{d_D \mid \exists r.D \in \mathrm{sub}(C) \cup \mathrm{sub}(\mathcal{T})\}$
- $A^{\mathcal{I}_{C,\mathcal{T}}} = \{d_D \mid D \sqsubseteq_{\mathcal{T}} A\}$, and
- $r^{\mathcal{I}_{C,\mathcal{T}}} = \{(d_D, d_E) \mid D \sqsubseteq_{\mathcal{T}} \exists r.E\}$.

The *canonical model $\mathcal{I}_{\mathcal{K}}$ of the knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$* consists of the domain $\Delta^{\mathcal{I}_{\mathcal{K}}}$ and the interpretation function $\cdot^{\mathcal{I}_{\mathcal{K}}}$ defined as follows:

- $\Delta^{\mathcal{I}_\mathcal{K}} = \{d_a \mid a \in \mathrm{Sig}_I\mathcal{A}\} \cup \{d_C \mid \exists r.C \in \mathrm{sub}(\mathcal{A}) \cup \mathrm{sub}(\mathcal{T})\}$,
- $A^{\mathcal{I}_\mathcal{K}} = \{d_D \mid D \sqsubseteq_\mathcal{T} A\} \cup \{d_a \mid \mathcal{K} \models A(a)$,
- $r^{\mathcal{I}_\mathcal{K}} = \{(d_D, d_E) \mid D \sqsubseteq_\mathcal{T} \exists r.E\} \cup \{(d_a, d_D) \mid \mathcal{K} \models \exists r.D(a)\} \cup$
  $\{(d_a, d_b) \mid r(a, b) \in \mathcal{A}\}$.

The canonical model $\mathcal{I}_{C,\mathcal{T}}$ is in some sense the most general model for the concept description C w.r.t. $\mathcal{T}$, as any other model $\mathcal{J}$ of $\mathcal{T}$ with an element $d \in C^\mathcal{J}$ can be simulated by the element d_C in $\mathcal{I}_{C,\mathcal{T}}$. Similarly, any other model $\mathcal{J}$ of $\mathcal{T}$ and an ABox $\mathcal{A}$ with $d = a^\mathcal{J}$ for an individual a is simulated by the element d_a in the canonical model $\mathcal{I}_\mathcal{K}$.

Theorem 5 (from [10])**.** *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, C and D be $\mathcal{EL}$-concept descriptions. Then:*

1. *for all models $\mathcal{I}$ of $\mathcal{T}$ and all elements $d \in \Delta^\mathcal{I}$ holds $d \in C^\mathcal{I}$ iff $(\mathcal{I}_{C,\mathcal{T}}, d_C) \lesssim (\mathcal{I}, d)$; and*
2. *$C \sqsubseteq_\mathcal{T} D$ iff $d_C \in D^{\mathcal{I}_{C,\mathcal{T}}}$*
   (or equivalently: $D \in \mathfrak{C}(\mathcal{I}_{C,\mathcal{T}}, d_C)$) iff $(\mathcal{I}_{D,\mathcal{T}}, d_D) \lesssim (\mathcal{I}_{C,\mathcal{T}}, d_C)$.

Note that canonical models for $\mathcal{EL}$-knowledge bases are always finite.

2.3 Concept Similarity Measures

A concept similarity measure for a DL $\mathcal{L}$ w.r.t. a TBox $\mathcal{T}$ is a function $\sim_\mathfrak{c}$: $\mathfrak{C}(\mathcal{L}) \times \mathfrak{C}(\mathcal{L}) \to [0, 1]$ that assigns to each pair of $\mathcal{L}$-concept descriptions a similarity value from the unit interval. A value $C \sim_\mathfrak{c} D = 0$ means that the concepts C and D are totally dissimilar, while a value of 1 denotes total similarity. For any concept description $C \in \mathfrak{C}(\mathcal{L})$, $C \sim_\mathfrak{c} C$ needs to be 1 for $\sim_\mathfrak{c}$ to be a well-defined measure. However, concept similarity measures can have additional properties, which could be useful depending on the application in which they are used. In particular, a concept similarity measure $\sim_\mathfrak{c}$ is:

- *symmetric*, iff $C \sim_\mathfrak{c} D = D \sim_\mathfrak{c} C$ for all $C, D \in \mathfrak{C}(\mathcal{L})$;
- *equivalence invariant*, iff for all $C, D, E \in \mathfrak{C}(\mathcal{L})$ with $C \equiv_\mathcal{T} D$ it holds that $C \sim_\mathfrak{c} E = D \sim_\mathfrak{c} E$;
- *equivalence closed*, iff $C \equiv_\mathcal{T} D \iff C \sim_\mathfrak{c} D = 1$ for all $C, D \in \mathfrak{C}(\mathcal{L})$;

- *bounded*, iff the existence of $E \neq \top$ with $C \sqsubseteq_{\mathcal{T}} E$ and $D \sqsubseteq_{\mathcal{T}} E$ implies $C \sim_{\mathfrak{c}} D > 0$ for all $C, D \in \mathfrak{C}(\mathcal{L})$;
- *dissimilar closed*, iff $C, D \neq \top$ and there is no $E \neq \top$ with $C \sqsubseteq_{\mathcal{T}} E$ and $D \sqsubseteq_{\mathcal{T}} E$ imply that $C \sim_{\mathfrak{c}} D = 0$ for all $C, D \in \mathfrak{C}(\mathcal{L})$;
- *subsumption preserving*, iff $C \sqsubseteq_{\mathcal{T}} D \sqsubseteq_{\mathcal{T}} E$ implies $C \sim_{\mathfrak{c}} D \geq C \sim_{\mathfrak{c}} E$ for all $C, D, E \in \mathfrak{C}(\mathcal{L})$;
- *reverse subsumption preserving*, iff $C \sqsubseteq_{\mathcal{T}} D \sqsubseteq_{\mathcal{T}} E$ implies $D \sim_{\mathfrak{c}} E \geq C \sim_{\mathfrak{c}} E$ for all $C, D, E \in \mathfrak{C}(\mathcal{L})$; and

These properties were listed and investigated for unfoldable $\mathcal{EL}$-TBoxes in [9]. However, for general TBoxes some properties need to be adapted to subsumption w.r.t. a TBox. In case of the properties bounded and dissimilar closed one could also have required equivalence (w.r.t. $\mathcal{T}$) to $\top$ instead of being syntactic equal. We chose the syntactic option, since in our opinion, if two concepts share a feature (even if it is equivalent to top), this still means they have something in common and should have a similarity value greater than 0. To guarantee formal properties as the ones above help making concept similarity measures more predictable and therefore more useful. Concept similarity measures should be parameterizable depending on the application domain, as often some properties of concepts are much more important than others for assessing the similarity. Furthermore, users often have an goal how the similarity between some particular concept pairs should be counted. The parameterizable similarity measures that we develop in the following sections allow ontology users to adapt the CSM to fit their expectations, while keeping (most of) the above properties.

3 Interpretation Similarity Measures

To define a concept similarity measure that works on general $\mathcal{EL}$-TBoxes, we first consider similarity measures on pointed interpretations. In general, an *interpretation similarity measure* is defined as a function of the type $\mathfrak{P} \times \mathfrak{P} \to [0, 1]$. It maps any pair of pointed interpretations to a similarity value between 0 and 1. We denote interpretation similarity measures by $\sim_{\mathfrak{P}}$.

There are various formal properties that interpretation similarity measures can have. Most of these transfer directly from the properties of concept similarity measures introduced by before. Given a DL $\mathcal{L}$, we define

$$\mathfrak{C}_{\mathcal{L}}((\mathcal{I}, d)) = \{C \in \mathfrak{C}(\mathcal{L}) \mid d \in C^{\mathcal{I}}\}$$

as the set of all $\mathcal{L}$-concept descriptions that have d as an element in $\mathcal{I}$. Given suitable relations on pointed interpretations $\lesssim_{\mathcal{L}}$ and $\simeq_{\mathcal{L}}$ that characterize subsumption and concept equivalence (like those from Definition 2 for $\mathcal{EL}$), we call an interpretation similarity measure:

- *symmetric*, iff $p \sim_i q = q \sim_i p$ for all $p, q \in \mathfrak{P}$;
- *bounded*, iff $\mathfrak{C}_{\mathcal{L}}(p) \cap \mathfrak{C}_{\mathcal{L}}(q) \supset \{\top\}$ implies $p \sim_i q > 0$ for all $p, q \in \mathfrak{P}$;
- *dissimilar closed*, iff $\mathfrak{C}_{\mathcal{L}}(p) \cap \mathfrak{C}_{\mathcal{L}}(q) = \{\top\}$ implies $p \sim_i q = 0$ for all $p, q \in \mathfrak{P}$ with $\mathfrak{C}_{\mathcal{L}}(p) \supset \{\top\}$ and $\mathfrak{C}_{\mathcal{L}}(q) \supset \{\top\}$;
- *equisimulation invariant*, iff $p \simeq_{\mathcal{L}} q$ implies $(p \sim_i u) = (q \sim_i u)$ for all $p, q, u \in \mathfrak{P}$;
- *equisimulation closed*, iff $p \simeq_{\mathcal{L}} q \iff p \sim_i q = 1$ for all $p, q \in \mathfrak{P}$;
- *simulation preserving*, iff $r \lesssim_{\mathcal{L}} q \lesssim_{\mathcal{L}} p$ implies $(p \sim_i q) \geq (p \sim_i r)$ for all $p, q, r \in \mathfrak{P}$;
- *reverse simulation preserving*, iff $r \lesssim_{\mathcal{L}} q \lesssim_{\mathcal{L}} p$ implies $(q \sim_i r) \geq (p \sim_i r)$ for all $p, q, r \in \mathfrak{P}$.

The parameterizable ISM that we develop in the following has most of these properties.

3.1 The Similarity Measure $\sim_i$ for Finite Interpretations

In this section, we define a family of interpretation similarity measures $\sim_i$ with a number of useful properties. Note that these properties are shown only for the case of the simulation relations defined in Definition 2, which correspond to subsumption and equivalence in $\mathcal{EL}$. This is important when lifting those properties to concept similarity measures (see Section 4.1). This interpretation similarity measure only works on *finite* interpretations. Instead of explicitly stating this over and over, we assume that all interpretations are finite for the rest of this section.

Given a pointed interpretation $(\mathcal{I}, d)$, we denote with

$$\mathrm{CN}((\mathcal{I}, d)) = \begin{cases} \{\top\} & \text{if there is no } A \in N_C \text{ with } d \in A^{\mathcal{I}} \\ \{A \in N_C \mid d \in A^{\mathcal{I}}\} & \text{otherwise} \end{cases}$$

$$\mathrm{SC}((\mathcal{I}, d)) = \left\{(r, (\mathcal{I}, e)) \in N_R \times \{(\mathcal{I}, e) \mid e \in \Delta^{\mathcal{I}}\} \mid (d, e) \in r^{\mathcal{I}}\right\}$$

the *set of concept names* that d is an instance of in $\mathcal{I}$, and the *set of direct successors* of d in $\mathcal{I}$, i.e., pairs of a role r and a pointed interpretation $q = (\mathcal{I}, e)$ such that d and e are connected in $\mathcal{I}$ by an edge labeled with r. Note that $\mathrm{CN}(p)$ is never empty, even if p is not an instance of any concept name, while $\mathrm{SC}(p)$ may be empty if p has no outgoing edges.

For two pointed interpretations to be perfectly similar, their designated elements need to have the same set of concept names and edges labeled with the same roles going to perfectly similar successor elements. Otherwise, the most similar concept names and the most similar direct successors are compared and a similarity value is computed from this. For both cases, we need the notion of pairings:

A pairing $P \subseteq X \times Y$ is a total binary relation, where totality means that all elements of X and all elements of Y appear in some tuple of P as the first component or second component, respectively. For two pointed interpretations p and q, we are interested in two types of pairings: the *concept name pairing* $P_C(p, q) \subseteq \mathrm{CN}(p) \times \mathrm{CN}(q)$ on the concept names that p and q are instances of; and the *successor pairing* $P_S(p, q)$ on the direct successors $\mathrm{SC}(p)$ and $\mathrm{SC}(q)$ of those elements. Thus the concept name pairing $P_C(p, q)$ always has at least one element. Recall that $\mathrm{SC}(p)$ may be empty. If in this case $\mathrm{SC}(q)$ is not empty, then the pairing P_S cannot be built. In this case, we introduce a new successor, using a new role $r_\top$ not appearing in $\mathcal{T}$ and that is dissimilar to all other role names. Thus, we define P_S as follows:

$$P_S(p, q) \subseteq \begin{cases} \mathrm{SC}(p) \times \mathrm{SC}(q) & \text{if } \mathrm{SC}(p) \neq \emptyset \wedge \mathrm{SC}(q) \neq \emptyset \\ \mathrm{SC}(p) \times \{(r_\top, q)\} & \text{if } \mathrm{SC}(p) \neq \emptyset \wedge \mathrm{SC}(q) = \emptyset \\ \{(r_\top, p)\} \times \mathrm{SC}(q) & \text{if } \mathrm{SC}(p) = \emptyset \wedge \mathrm{SC}(q) \neq \emptyset \\ \emptyset & \text{if } \mathrm{SC}(p) = \emptyset \wedge \mathrm{SC}(q) = \emptyset \end{cases}$$

The successor pairing $P_S(p, q)$ may be empty if both p and q have no successors. However, as soon as one of p or q is instance of a concept name or has a direct successor, this concept name or successor appears in the pairing P_C or P_S, since those pairings are always total.

The interpretation similarity measure $\sim_i$ is defined based on a primitive measure, i.e. a measure that assigns similarity values to each pair of basic concepts (i.e., concept names or $\top$) and each pair of role names

Definition 6. A *primitive measure* is a function

$$\sim_{\mathrm{prim}} : (N_C \cup \{\top\}) \times (N_C \cup \{\top\}) \cup N_R \times N_R \to [0, 1]$$

that satisfies the following properties:

- $x \sim_{\text{prim}} x = 1$ for any role name or basic concept x,
- $\top \sim_{\text{prim}} A = A \sim_{\text{prim}} \top = 0$ for all $A \in N_C$, and
- $r_\top \sim_{\text{prim}} s = s \sim_{\text{prim}} r_\top = 0$ for all role names $s \neq r_\top$.

Additionally, in order to obtain a symmetric similarity measure $\sim_i$, $\sim_{\text{prim}}$ needs to be symmetric as well.

We introduce a default primitive measure, that simply assigns similarity 0 to pairs of different basic concepts or role names:

$$x \sim_{\text{default}} y = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

However, other primitive measures could be useful as well. For example, one might want to express that two colors Red and Orange are similar to *some* degree even if they are modelled by different concept names.

Additionally, one can assign weights to different basic concepts and role names using a weighting function $g : N_C \cup \{\top\} \cup N_R \to \mathbb{R}_{>0}$ to prioritize different features in the similarity measure. This function g is extended to pairs of basic concepts or role names as $g(A, B) = \max(g(A), g(B))$ and $g(r, s) = \max(g(r), g(s))$; this means, if we compute the primitive similarity between two basic concepts A and B (because those occur in the pairing P_C), this value is multiplied with the maximum weight of $g(A)$ and $g(B)$, and analogously for roles.

Finally, any primitive measure $\sim_{\text{prim}}$ and weighting function g can be extended to a similarity measure on pointed interpretations by recursively traversing the interpretation graphs, computing the primitive measure for the best concept name and successor pairing at each element:

$$p \sim_i q = \max_{\substack{P_C(p,q) \\ P_S(p,q)}} \left(\frac{\text{sim}(P_C) + \text{sim}(P_S)}{\sum\limits_{(A,B) \in P_C} g(A, B) + \sum\limits_{((r,p'),(s,q')) \in P_S} g(r, s)} \right) \tag{1}$$

where

$$\begin{aligned} \text{sim}(P_C) &= \sum_{(A,B) \in P_C} g(A, B)(A \sim_{\text{prim}} B) \\ \text{sim}(P_S) &= \sum_{((r,p'),(s,q')) \in P_S} g(r, s)(r \sim_{\text{prim}} s)((1 - w) + w(p' \sim_i q')) \end{aligned}$$

The constant w allows for discounting of successors, and has a value $w \in (0, 1)$. Figure 1 shows an example of which successors might be chosen by a successor

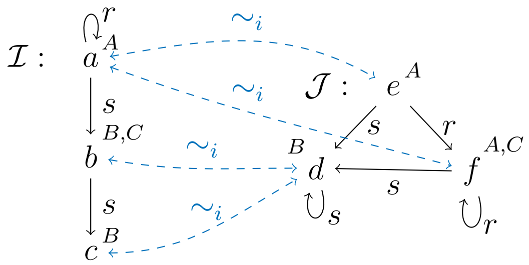

Figure 1: Recursive computation of the similarity value between $(\mathcal{I}, a)$ and $(\mathcal{J}, e)$.

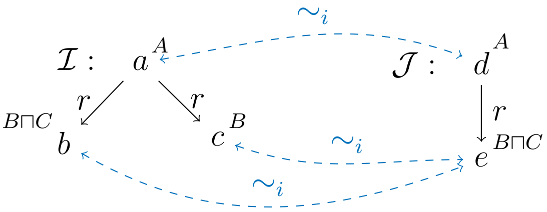

Figure 2: Two pointed interpretations with $(\mathcal{I}, a) \simeq (\mathcal{J}, d)$, where the naive application of the similarity measure $\sim_i$ yields similarity values less than 1.

pairing to compute the maximal similarity value. Both designated elements a and e have an r- and s-successor, for which the similarity value needs to computed as well. The s-successors have s-successors on their own, yielding all pairs of pointed interpretations over $\mathcal{I}$ and $\mathcal{J}$, respectively, whose similarities affect the similarity between a and e.

Note that by defining the similarity measure this way, it is not equisimulation closed. The reason is that the successor pairing always connects successors symmetrically, which gives rise to problems for sibling successor nodes that are in a subsumption relationship. For example, consider the pointed interpretations in Figure 2. For these pointed interpretations, any complete successor pairing must map the node c as a successor of a to the only successor of d: the node e. The similarity between c and e is less than 1 and affects the resulting similarity $(\mathcal{I}, a) \sim_i (\mathcal{J}, d)$, which might get also a value less than 1, even though $(\mathcal{I}, a)$ and $(\mathcal{J}, d)$ clearly simulate each other.

There are multiple solutions to this problem: First, one can modify the similarity measure to be more in line with the notion of simulations by computing a directional similarity between nodes and always return value 1 if all concept names and successors from one node occur in the other node (and maybe even more). However, this would complicate the whole algorithm. Instead we take a similar approach to [9] and normalize the interpretations $\mathcal{I}$ and $\mathcal{J}$ before applying the similarity measure.

Definition 7 (normal form for interpretations)**.** An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is in normal form if there are no elements $a, b, c \in \Delta^{\mathcal{I}}$ with $\{(a, b), (a, c)\} \subseteq r^{\mathcal{I}}$ and $(\mathcal{I}, b) \lesssim (\mathcal{I}, c)$, i.e., no node has two successor nodes for the same role name that are in a simulation relation.

Any interpretation $\mathcal{I}$ can be transformed into normal form as follows:

1. remove all edges $(a, b) \in r^{\mathcal{I}}$ in the interpretation graph, for which there exists an edge $(a, c) \in r^{\mathcal{I}}$ with $(\mathcal{I}, b) \lesssim (\mathcal{I}, c)$ and $(\mathcal{I}, b) \not\simeq (\mathcal{I}, c)$
2. for all edges $(a, b_0) \in r^{\mathcal{I}}$, check if there are other edges $(a, b_i) \in r^{\mathcal{I}}$, $i > 0$, with $(\mathcal{I}, b_0) \simeq (\mathcal{I}, b_i)$ and choose one representative b_j; then remove all other edges (a, b_i), $i \neq j$, from $r^{\mathcal{I}}$.

Note that this normalization is well-defined, since we assume that the pointed interpretations are always finite, and simulations can be computed in polynomial time in the size of the interpretation [7]. If we always normalize the pointed interpretations in a preprocessing step, we can show that the similarity measure $\sim_i$ is now equivalence closed. This is a direct consequence of the following lemma.

Lemma 8. *Let $(\mathcal{I}, a)$ and $(\mathcal{J}, b)$ be two pointed interpretations and let $\mathcal{I}'$ and $\mathcal{J}'$ be the results of normalizing $\mathcal{I}$ and $\mathcal{J}$, respectively. Then the following holds:*

1. *Normalization preserves simulations, i.e., if $(\mathcal{I}, a) \lesssim (\mathcal{J}, b)$ then also $(\mathcal{I}', a) \lesssim (\mathcal{J}', b)$.*
2. *If $(\mathcal{I}, a) \simeq (\mathcal{J}, b)$, then for any successor $(r, p) \in \mathrm{SC}((\mathcal{I}', a))$ there exists a unique successor $(r, q) \in \mathrm{SC}((\mathcal{J}', b))$ with $p \simeq q$ and vice versa. We denote this property by saying that $(\mathcal{I}', a)$ and $(\mathcal{J}', b)$ are* structurally equivalent.

Proof.

1. Let $(\mathcal{I}, a)$ and $(\mathcal{J}, b)$ be two pointed interpretations with $(\mathcal{I}, a) \lesssim (\mathcal{J}, b)$. Then for each concept name A, we have $a \in A^{\mathcal{I}'} \Leftrightarrow a \in A^{\mathcal{I}} \Rightarrow b \in A^{\mathcal{J}} \Leftrightarrow b \in A^{\mathcal{J}'}$. Additionally, for each role name r, we have $(a, a') \in r^{\mathcal{I}'} \Rightarrow (a, a') \in r^{\mathcal{I}} \Rightarrow \exists b' : (b, b') \in r^{\mathcal{J}} \wedge (\mathcal{I}, a') \lesssim (\mathcal{J}, b')$. If $(b, b') \in r^{\mathcal{J}'}$, we are done: $(\mathcal{I}', a) \lesssim (\mathcal{J}', b)$ follows directly.

   Otherwise, we know by the construction of $\mathcal{J}'$, that there exists an element $c \in \Delta^{\mathcal{J}'}$ with $(b, c) \in r^{\mathcal{J}'}$ and $(\mathcal{J}', b') \lesssim (\mathcal{J}', c)$ or $(\mathcal{J}', b') \simeq (\mathcal{J}', c)$. Since $\lesssim$ is transitive and $(\mathcal{I}', a') \lesssim (\mathcal{I}, a) \lesssim (\mathcal{J}, c)$, this means that $(\mathcal{I}', a') \lesssim (\mathcal{J}', c)$ and the claim, $(\mathcal{I}', a) \lesssim (\mathcal{J}', b)$ again follows.

2. Let $(\mathcal{I}, a)$ and $(\mathcal{J}, b)$ be two pointed interpretations with $(\mathcal{I}, a) \simeq (\mathcal{J}, b)$. Let further $(a, c) \in r^{\mathcal{I}'}$, which also implies $(a, c) \in r^{\mathcal{I}}$. Since $\mathcal{I}'$ is in normal form, this means that there is no $c' \in \Delta^{\mathcal{I}}$ with $(a, c') \in r^{\mathcal{I}}$ and $(\mathcal{I}, c) \lesssim (\mathcal{I}, c')$, and $(\mathcal{I}, c') \not\lesssim (\mathcal{I}, c)$. Since $(\mathcal{I}, a) \simeq (\mathcal{J}, b)$, there exists an element $d \in \Delta^{\mathcal{J}}$ with $(b, d) \in r^{\mathcal{J}}$ and $(\mathcal{I}, c) \lesssim (\mathcal{J}, d)$, but not necessarily $(b, d) \in r^{\mathcal{J}'}$. By the construction of $\mathcal{J}'$, we know that there is an element $e \in \Delta^{\mathcal{J}'}$ with $(b, e) \in r^{\mathcal{J}'}$ and $(\mathcal{J}, d) \lesssim (\mathcal{J}, e)$. Again, $(\mathcal{I}, a) \simeq (\mathcal{J}, b)$ implies that a must have a successor $(a, f) \in r^{\mathcal{I}}$ with $(\mathcal{J}, e) \lesssim (\mathcal{I}, f)$; however, since with $(\mathcal{I}, c) \lesssim (\mathcal{J}, d)$ and $(\mathcal{J}, d) \lesssim (\mathcal{J}, e)$, this also means $(\mathcal{I}, c) \lesssim (\mathcal{I}, f)$. Since we assumed that there is no $c' \in \Delta^{\mathcal{I}}$ with $(a, c') \in r^{\mathcal{I}}$ and $(\mathcal{I}, c) \lesssim (\mathcal{I}, c')$, this means that $f = c$ and thus $(\mathcal{I}, c) \simeq (\mathcal{J}, e)$ and by point 1. also $(\mathcal{I}', c) \simeq (\mathcal{J}', e)$. The other direction is analogous. □

In the following, whenever we write $(\mathcal{I}, a) \sim_i (\mathcal{J}, b)$, we implicitly assume that $\mathcal{I}$ and $\mathcal{J}$ have been normalized first.

3.2 Properties of the ISM $\sim_i$

Before discussing the properties of the similarity measure $\sim_i$, we first show that it is actually well-defined, even for cyclic interpretations.

Lemma 9. *The similarity measure $\sim_i$ is well-defined, i.e., Equation* (1) *has a unique solution for any two pointed interpretations $(\mathcal{I}, a)$ and $(\mathcal{J}, b)$.*

Proof. If we fix the two interpretations $\mathcal{I}$ and $\mathcal{J}$, we can view $\sim_i$ as an iterative function that "refines" the similarities between any two elements $(c, d) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$, i.e., a function on the vector space $\mathbb{R}^{|\Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}|}$. In particular, since the value of $p' \sim_i q'$ in Equation 1 is always multiplied with w (there may be other factors, which are always less than 1), $\sim_i$ is Lipschitz continuous with a Lipschitz constant of at most w. Because $w < 1$, this means that $\sim_i$ is a contraction mapping on $\mathbb{R}^{|\Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}|}$. But then, the Banach fixed-point theorem implies that $\sim_i$ has a unique fixed point in $\mathbb{R}^{|\Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}|}$, and indeed the iteration of $\sim_i$ on any starting tuple (like starting with a similarity of 0 between any pair of elements) converges to this fixed point [3]. This unique fixed-point means that Equation (1) has a unique solution for any $(\mathcal{I}, a) \sim_i (\mathcal{J}, b)$ (which corresponds exactly to the value between a and b for the fixed point) and is thus well-defined. □

Even though $\sim_i$ as given in Equation (1) is well-defined, it cannot be used directly to compute the similarity value, since cycles in the interpretation would lead to infinite recursion. Instead, we will give an iterative algorithm to compute the similarities between all elements of two interpretations $\mathcal{I}$ and $\mathcal{J}$, whose values converge to the fixed point. By iterating this algorithm a number of times, e.g. until a given tolerance or a maximum number of iterations is reached, the exact similarity can be approximated arbitrarily close. The correctness of this algorithm follows again from the Banach fixed-point theorem given above, as does the convergence factor of w:

- $(\mathcal{I}, d) \sim_i^0 (\mathcal{J}, e) = 0$ for all $d \in \Delta^\mathcal{I}$ and $e \in \Delta^\mathcal{J}$;
- $(\mathcal{I}, d) \sim_i^{n+1} (\mathcal{J}, e) = \max\limits_{\substack{P_C((\mathcal{I},d),(\mathcal{J},e))\\ P_S((\mathcal{I},d),(\mathcal{J},e))}} \left(\dfrac{\mathrm{sim}(P_C) + \mathrm{sim}(P_S)}{\sum\limits_{(A,B)\in P_C} g(A,B) + \sum\limits_{((r,p'),(s,q'))\in P_S} g(r,s)} \right)$
  for all $d \in \Delta^\mathcal{I}$ and $e \in \Delta^\mathcal{J}$, where:
  $$\mathrm{sim}(P_C) = \sum_{(A,B)\in P_C} g(A,B)(A \sim_{\mathrm{prim}} B)$$
  $$\mathrm{sim}(P_S) = \sum_{((r,p'),(s,q'))\in P_S} g(r,s)(r \sim_{\mathrm{prim}} s)((1-w) + w(p' \sim_i^n q'))$$

We now show that this similarity measure has many of the nice properties given at the beginning of this chapter.

Theorem 10. *Let $\sim_i$ be an ISM instantiated with a constant $w \in (0,1)$, a weighting function g and a primitive measure $\sim_{\mathrm{prim}}$. Then $\sim_i$ is bounded and equisimulation invariant for $\mathcal{EL}$-concepts defined w.r.t. an $\mathcal{EL}$-TBox and $\lesssim$.*

Proof. 1. *bounded:* $\sim_i$ is bounded, if $\mathfrak{C}(p) \cap \mathfrak{C}(q) \supset \{\top\}$ implies $p \sim_i q > 0$ for all $p, q \in \mathfrak{P}$. Assume that there exists a concept $C \neq \top$ in $\mathfrak{C}(p) \cap \mathfrak{C}(q)$. Then, there also exists either a concept name A or an existential restriction of the form $\exists r.\top$ in $\mathfrak{C}(p) \cap \mathfrak{C}(q)$, since for all conjunctions $C_1 \sqcap C_2 \in \mathfrak{C}(p) \cap \mathfrak{C}(q)$ we also have $C_1, C_2 \in \mathfrak{C}(p) \cap \mathfrak{C}(q)$ and for all $\exists r.C \in \mathfrak{C}(p) \cap \mathfrak{C}(q)$ we also have $\exists r.\top \in \mathfrak{C}(p) \cap \mathfrak{C}(q)$.

However, for a concept name $A \in \mathfrak{C}(p) \cap \mathfrak{C}(q)$, we have that $A \sim_{\mathrm{prim}} A = 1$ and thus $\max_{P_C(p,q)} \sum_{(A,B)\in P_C} g(A,B)(A \sim_{\mathrm{prim}} B) > 0$. This yields $p \sim_i q > 0$. Correspondingly, for $\exists r.\top \in \mathfrak{C}(p) \cap \mathfrak{C}(q)$, we have $1 = (r \sim_{\mathrm{prim}} r)$ and thus $g(r,s)(r \sim_{\mathrm{prim}} r)((1-w) + w(p' \sim_i q')) > 1 - w > 0$ and $\max_{P_S(p,q)} \sum_{((r,p'),(s,q'))\in P_S} g(r,s)(r \sim_{\mathrm{prim}} s)((1-w)+w(p' \sim_n q')) > 0$. Again, this yields $p \sim_i q > 0$.

2. *equisimulation invariant:* $\sim_i$ is equisimulation invariant, if $p \simeq q$ implies $p \sim_i u = q \sim_i u$ for all $p, q, u \in \mathfrak{P}$; This is a direct consequence of the fact that if $p \simeq q$, then the normalized pointed interpretations do not just simulate each other, but are structurally equivalent, as stated in point 2 in Lemma 8.

□

The parameterizable ISM $\sim_i$ needs to be instantiated appropriately to enjoy some of the remaining properties.

Theorem 11. *For $\mathcal{EL}$-concepts defined w.r.t. an $\mathcal{EL}$-TBox and $\lesssim$ the ISM $\sim_i$ is*

- *symmetric, if $\sim_i$ is instantiated with a constant $w \in (0,1)$, a weighting function g and a symmetric primitive measure $\sim_{\text{prim}}$.*
- *dissimilar closed, if $\sim_i$ is instantiated with a constant $w \in (0,1)$, a weighting function g and a primitive measure $\sim_{\text{prim}}$ that does not assign a similarity value greater than 0 to different concept or role names.*
- *equisimulation closed, if $\sim_i$ is instantiated with a constant $w \in (0,1)$, a weighting function g and a primitive measure $\sim_{\text{prim}}$ that does not assign the similarity value 1 to different concept or role names.*

Proof. 1. *symmetric:* $\sim_i$ is symmetric, if the primitive measure $\sim_{\text{prim}}$ is symmetric, as the definition of $\sim_i$ only uses commutative operators.

2. *dissimilar closed:* $\sim_i$ is dissimilar closed, if $\mathfrak{C}(p) \cap \mathfrak{C}(q) = \{\top\}$ implies $p \sim_i q = 0$ for all $p, q \in \mathfrak{P}$ with $\mathfrak{C}(p) \supset \{\top\}$ and $\mathfrak{C}(q) \supset \{\top\}$; of course, $\sim_i$ can only be dissimilarity closed if the primitive measure does not assign a similarity value greater than 0 to different concept or role names. Hence we only show this property for the default primitive measure $\sim_{\text{default}}$.

   Let $p, q \in \mathfrak{P}$ with $\mathfrak{C}(p) \supset \{\top\}$ and $\mathfrak{C}(q) \supset \{\top\}$, i.e., both p and q are instance of some concept name or have a successor. If $\mathfrak{C}(p) \cap \mathfrak{C}(q) = \{\top\}$, then $A \sim_{\text{default}} B = 0$ for all $A \in \text{CN}(p)$ and $B \in \text{CN}(q)$. Similarly, as there is no role name r with $(r, p') \in \text{SC}(p)$ and $(r, q') \in \text{SC}(q)$, we have $r \sim_{\text{default}} s = 0$ for all $(r, p') \in S(p)$ and $(s, q') \in S(q)$. This then yields $p \sim_i q = 0$.

3. *equisimulation closed:* The direction from left to right, i.e., $p \simeq q$ implies $p \sim_i q = 1$, follows again by point 2 in Lemma 8. For the other direction, that $p \sim_i q = 1$ also implies $p \simeq q$, we need the property that the primitive

measure does not assign a similarity value of 1 to different concept or role names. In this case, assume that $p \not\simeq q$ for $p = (\mathcal{I}, a)$ and $q = (\mathcal{J}, b)$. Then, w.l.o.g., we will have one of the following conditions:

(a) there exists a concept name A with $a \in A^{\mathcal{I}}$ and $b \notin A^{\mathcal{J}}$, or

(b) a has a successor $(a, c) \in r^{\mathcal{I}}$ and there is no d with $(b, d) \in r^{\mathcal{J}}$, or

(c) a has a successor $(a, c) \in r^{\mathcal{I}}$ and for all successors $t = (\mathcal{J}, d)$ of b with $(b, d) \in r^{\mathcal{J}}$ we have that $s \not\simeq t$. In this case, there must be a finite chain of such successors s_i, t_i starting from a, b such that Condition 1 or 2 holds for s_n, t_n.

Now, we can prove inductively that $p \sim_i q < 1$. In the first two cases a) and b), Equation 1 directly gives a similarity value < 1, since the concept name A in case a) or the role name r in case b) will always be matched with a different concept or role name and $\sim_{\text{prim}}$ never assigns similarity 1 to this match. In the third case, we assume that $c \sim_i d < 1$ by induction for all successors d of b. Then Equation 1 again yields a similarity value $p \sim_i q < 1$. Thus $\sim_i$ must equisimulation closed. □

Observe that $\sim_i$ can be used for two pointed interpretations that do not only differ in the designated elements, but that differ in the underlying interpretations. This allows to employ $\sim_i$ to compare different finite models and to assess their similarity. Thus $\sim_i$ can be used for ontology alignment for $\mathcal{EL}$-TBoxes, where named concepts defined over one TBox are mapped to a corresponding named concept in another TBox. One can compute the maximally similar pairs of nodes of both canonical models for a domain specific instantiation of $\sim_i$.

However, in this paper we want to use $\sim_i$ to derive a CSM that enables us to devise a computation algorithm for relaxed instances in $\mathcal{EL}$.

4 Relaxing Instance Queries using $\sim_c$

Relaxed instance queries based on CSMs were recently introduced in [5]. They generalize the notion of instance queries for a given query concept C by returning not only the exact instances of C, but also individuals that are similar enough, i.e., being an instance of a concept that has a similarity greater than a given threshold to the query concept C w.r.t. a given CSM.

4.1 The Concept Similarity Measure $\sim_c$

This type of inference service requires a similarity measure on concepts, since it also includes a generalization: For an individual a to be a relaxed instance of C, we require that the maximal similarity between the query concept and all of the concepts that have a as an instance is larger than t. This notion cannot be captured easily by interpretation similarity measures. We modify the interpretation similarity measure $\sim_i$ to work on concept descriptions and employ canonical models as means to translate arbitrary concept descriptions with respect to background knowledge into pointed interpretations.

Using canonical models, we define a concept similarity measure $\sim_c$ on $\mathcal{EL}$-concept descriptions w.r.t. a general $\mathcal{EL}$-TBox $\mathcal{T}$ as follows:

$$C \sim_c D = (\mathcal{I}_{C,\mathcal{T}}, d_C) \sim_i (\mathcal{I}_{D,\mathcal{T}}, d_D).$$

The concept similarity measure $\sim_c$ inherits the nice properties of $\sim_i$, since the properties for interpretation similarity measures were defined to correspond exactly to the concept similarity properties given in Section 2.3.

Theorem 12 (Properties of $\sim_c$)**.** *The concept similarity measure $\sim_c$ is symmetric, bounded, dissimilar closed, equivalence invariant, and equivalence closed, if $\sim_i$ is symmetric, bounded dissimilar closed, equisimulation invariant and equisimulation closed, respectively.*

Proof. We prove that the properties of $\sim_i$ transfer to $\sim_c$:

1. symmetry: $C \sim_c D = (\mathcal{I}_{C,\mathcal{T}}, d_C) \sim_i (\mathcal{I}_{D,\mathcal{T}}, d_D) \overset{\text{symmetry of } \sim_i}{=} (\mathcal{I}_{D,\mathcal{T}}, d_D) \sim_i (\mathcal{I}_{C,\mathcal{T}}, d_C) = D \sim_c C$.

2. bounded: Assume that for two $\mathcal{EL}$-concept descriptions C and D, there exists a concept $E \neq \top$ with $C \sqsubseteq_{\mathcal{T}} E$ and $D \sqsubseteq_{\mathcal{T}} E$. Then Theorem 5 yields $E \in \mathfrak{C}(p) \cap \mathfrak{C}(q)$ for $p = (\mathcal{I}_{C,\mathcal{T}}, d_C)$ and $q = (\mathcal{I}_{D,\mathcal{T}}, d_D)$. Therefore boundedness of $\sim_i$ implies $C \sim_c D = p \sim_i q > 0$.

3. dissimilar closed: Assume that for two $\mathcal{EL}$-concept descriptions $C, D \neq \top$, there is no concept description $E \neq \top$ with $C \sqsubseteq_{\mathcal{T}} E$ and $D \sqsubseteq_{\mathcal{T}} E$. Then Theorem 5 implies that $\mathfrak{C}(p) \cap \mathfrak{C}(q) = \{\top\}$ for $p = (\mathcal{I}_{C,\mathcal{T}}, d_C)$ and $q = (\mathcal{I}_{D,\mathcal{T}}, d_D)$, and thus, since $\sim_i$ is dissimilar closed (and thus the primitive measure does not assign a similarity value greater than 0 to different concept or role names), $C \sim_c D = p \sim_i q = 0$.

4. equivalence invariant: Assume that $C \equiv_\mathcal{T} D$. Then by Theorem 5 we have $(\mathcal{I}_{C,\mathcal{T}}, d_C) \simeq (\mathcal{I}_{D,\mathcal{T}}, d_D)$ and thus equisimulation invariance of $\sim_i$ implies $(\mathcal{I}_{C,\mathcal{T}}, d_C) \sim_i (\mathcal{J}, e) = (\mathcal{I}_{D,\mathcal{T}}, d_D) \sim_i (\mathcal{J}, e)$ for any pointed interpretation $(\mathcal{J}, e)$, in particular pointed interpretations of the form $(\mathcal{I}_{E,\mathcal{T}}, d_E)$. This then yields $C \sim_c E = D \sim_c E$ for any $\mathcal{EL}$-concept description E.

5. equivalence closed: Assume that $C \equiv_\mathcal{T} D$. Then by Theorem 5 we have $(\mathcal{I}_{C,\mathcal{T}}, d_C) \simeq (\mathcal{I}_{D,\mathcal{T}}, d_D)$ and thus $(\mathcal{I}_{C,\mathcal{T}}, d_C) \sim_i (\mathcal{I}_{D,\mathcal{T}}, d_D) = 1$ since $\sim_i$ is equisimulation closed. But then we also have $C \sim_c D = 1$.

   Similarly, assume that $C \sim_c D = (\mathcal{I}_{C,\mathcal{T}}, d_C) \sim_i (\mathcal{I}_{D,\mathcal{T}}, d_D) = 1$. Then $(\mathcal{I}_{C,\mathcal{T}}, d_C) \simeq (\mathcal{I}_{D,\mathcal{T}}, d_D)$ since $\sim_i$ is equisimulation closed, and thus Theorem 5 yields $C \equiv_\mathcal{T} D$. □

4.2 Using $\sim_c$ for Relaxed Instance Queries

Concept similarity-based relaxed instance queries were first defined in [5].

Definition 13 (relaxed instance)**.** Let $\mathcal{L}$ be a DL, $\sim_\mathfrak{c}$ be a CSM, and $t \in [0, 1)$. The individual $a \in N_I$ is a *relaxed instance* of the query concept Q w.r.t. the $\mathcal{L}$-knowledge base $\mathcal{K}$, $\sim_\mathfrak{c}$ and the threshold t iff there exists a $\mathcal{L}$-concept description $X \in \mathfrak{C}(\mathcal{L})$ such that $Q \sim_\mathfrak{c} X > t$ and $\mathcal{K} \models X(a)$. With $\mathrm{Relax}_t^{\sim_\mathfrak{c}}(Q)$ we denote the set of all individuals occurring in $\mathcal{K}$ that are relaxed instances of Q w.r.t. $\mathcal{K}$, $\sim_\mathfrak{c}$ and t.

In order to devise a computation algorithm for this inference, we need to introduce some notions. The *generalized concepts* of a concept $C = \bigsqcap_{i\in I} A_i \sqcap \bigsqcap_{j\in J} \exists r_j.C_j$ are of the form $D = \bigsqcap_{i\in I'} A_i \sqcap \bigsqcap_{j\in J'} \exists r_j.D_j$ for $I' \subseteq I$, $J' \subseteq J$, and D_j are generalized concepts of C_j for all $j \in J'$. This means, generalized concepts of a concept description C are always obtained by deleting concept names or existential restrictions anywhere in C. A concept C is *fully expanded* w.r.t. the TBox $\mathcal{T}$ if any concept description $\exists r_1 \dots \exists r_n.E$ with $C \sqsubseteq_\mathcal{T} \exists r_1 \dots \exists r_n.D$ and there is a GCI $D \sqsubseteq E$ in $\mathcal{T}$ is equivalent to a generalized concept of C (i.e., C contains all its implications explicitly. It was also established in [5], that all concepts Q' that have a as an instance are equivalent to a generalized concept of the (possibly infinite [8]) fully expanded $\mathrm{msc}_\mathcal{K}(a)$. The fully expanded $\mathrm{msc}_\mathcal{K}(a)$ in $\mathcal{EL}$ is exactly the tree unraveling of $\mathcal{I}_\mathcal{K}$ starting from d_a (see [14]), and thus for any concept C we have

$$C \sim_c \mathrm{msc}_\mathcal{K}(a) = (\mathcal{I}'_{C,\mathcal{T}}, d_C) \sim_i (\mathcal{I}'_\mathcal{K}, d_a),$$

where $\mathcal{I}'_{C,\mathcal{T}}$ and $\mathcal{I}'_\mathcal{K}$ are the normalized canonical models of C and the TBox $\mathcal{T}$ or of the KB $\mathcal{K}$, respectively.

To compute the maximal similarity between the query concept Q and generalized concepts of the $\mathrm{msc}_{\mathcal{K}}(a)$ we can simply modify the definition of $\sim_i$ in Equation (1) on page 12 to check all subsets of the concept names $S_{\mathrm{CN}} \subseteq \mathrm{CN}(q)$ and successors $S_{\mathrm{SC}} \subseteq \mathrm{SC}(q)$ of the pointed interpretations q in the canonical model $\mathcal{I}_{\mathcal{K}}$ and maximize $\sim_i$ over those subsets. This corresponds to checking generalized concepts of the $\mathrm{msc}_{\mathcal{K}}(a)$.

Note however, that not all generalized concepts are checked, since the canonical models are always finite and using the subset construction, only finitely many generalized concepts can be created. Whereas the $\mathrm{msc}_{\mathcal{K}}(a)$ may be infinite and thus can have infinitely many generalized concepts. However, to find the maximal similarity, the above subset construction is sufficient, since any infinite $\mathrm{msc}_{\mathcal{K}}(a)$ is at some point cyclic, and thus we can reuse the same subsets for recurring elements (which correspond exactly to the same pair (p, q) of pointed interpretations).

The following Equations compute the maximal similarity between a pointed interpretation p and all 'generalized pointed interpretation' obtained from q by the subset construction.

$$p \sim_{\mathrm{imax}} q = \max_{\substack{S_{\mathrm{CN}} \subseteq \mathrm{CN}(q) \\ S_{\mathrm{SC}} \subseteq \mathrm{SC}(q)}} \max_{\substack{P_C \subseteq \mathrm{CN}(p) \times S_{\mathrm{CN}} \\ P_S \subseteq \mathrm{SC}(p) \times S_{\mathrm{SC}}}} \frac{\mathrm{sim}(P_C) + \mathrm{sim}(P_S)}{\sum_{(A,B) \in P_C} g(A,B) + \sum_{((r,p'),(s,q')) \in P_S} g(r,s)}$$

where

$$\begin{aligned} \mathrm{sim}(P_C) &= \sum_{(A,B) \in P_C} g(A,B)(A \sim_{\mathrm{prim}} B) \\ \mathrm{sim}(P_S) &= \sum_{((r,p'),(s,q')) \in P_S} g(r,s)(r \sim_{\mathrm{prim}} s)((1-w) + w \cdot (p' \sim_{\mathrm{imax}} q') \end{aligned}$$

A deterministic algorithm to compute relaxed instances for $\sim_c$ is given in Figure 3.

The maxsim_i values computed in the algorithm converge monotonically from below to the maximal similarities between generalized concepts of the most specific concept of an individual and the query concept. Thus, for any individual a, which is a relaxed instance of Q with a threshold strictly larger than t, there exists $i \in \mathbb{N}$ such that for all $j > i$ we have $\mathrm{maxsim}_j(Q, a) > t$. Thus, the algorithm is sound and complete in the following sense:

Theorem 14. *Let $\sim_c$ be the CSM derived from $\sim_i$ with the primitive measure $\sim_{\mathrm{prim}}$, the weighting function g and the discounting factor w. The algorithm* relaxed-instances *is sound and complete:*

1. Soundness*: If $a \in$ relaxed-instances$(Q, \mathcal{K}, t, \sim_{\mathrm{prim}}, g, w)$ for a number n of iterations, then $a \in \mathrm{Relax}_t^{\sim_c}(Q)$.*

```
Procedure: relaxed-instances $(Q, \mathcal{K}, t, \sim_{\text{prim}}, g, w)$
Input: $Q$: $\mathcal{EL}$-concept description; $\mathcal{K} = (\mathcal{T}, \mathcal{A})$: $\mathcal{EL}$-KB; $t \in [0,1]$: threshold;
    $\sim_{\text{prim}}$: primitive measure; $w \in (0,1)$: discounting factor
Output: individuals $a \in \mathit{Relax}_t^{\sim_c}(Q)$
compute canonical models $\mathcal{I}_{Q,\mathcal{T}}$ and $\mathcal{I}_{\mathcal{K}}$
normalize canonical models to $\mathcal{I}'_{Q,\mathcal{T}}$ and $\mathcal{I}'_{\mathcal{K}}$
$\text{maxsim}_0(d, e) \leftarrow 0$ for all $d \in \Delta^{\mathcal{I}'_{Q,\mathcal{T}}}$ and $e \in \Delta^{\mathcal{I}'_{\mathcal{K}}}$
for $i \leftarrow 1$ to $n$ do
    for all $d \in \Delta^{\mathcal{I}'_{Q,\mathcal{T}}}$ and $e \in \Delta^{\mathcal{I}'_{\mathcal{K}}}$ do
        $\text{maxsim}_i(d, e) \leftarrow \max_{\substack{S_{\text{CN}} \subseteq \text{CN}(e) \\ S_{\text{SC}} \subseteq \text{SC}(e)}} \max_{\substack{P_C \subseteq \text{CN}(d) \times S_{\text{CN}} \\ P_S \subseteq \text{SC}(d) \times S_{\text{SC}}}} \mathsf{similarity}(P_C, P_S, \sim_{\text{prim}}, g, w, i)$
    end for
end for
return $\{a \in \text{Sig}_I(\mathcal{A}) \mid \text{maxsim}_n(d_Q, d_a) \geq t\}$

Procedure: similarity $(P_C, P_S, \sim_{\text{prim}}, g, w, i)$
$\text{sim}(P_C) = \sum_{(A,B) \in P_C} g(A, B)(A \sim_{\text{prim}} B)$
$\text{sim}(P_S) = \sum_{((r,f),(s,g)) \in P_S} g(r, s)(r \sim_{\text{prim}} s)\big((1 - w) + w \cdot \text{maxsim}_{i-1}(f, g)\big)$
return $\dfrac{\text{sim}(P_C) + \text{sim}(P_S)}{\sum_{(A,B) \in P_C} g(A, B) + \sum_{((r,f),(s,g)) \in P_S} g(r, s)}$
```

Figure 3: Algorithm to compute all relaxed instances of a query concept Q w.r.t. the knowledge base $\mathcal{K}$, the threshold t and the similarity measure $\sim_c$ defined by the primitive measure $\sim_{\text{prim}}$, the weighting function g and the discounting factor w.

2. Completeness*: If* $a \in \text{Relax}_t^{\sim_c}(Q)$*, then there is a number* $n \in \mathbb{N}$ *such that* $a \in$ relaxed-instances$(Q, \mathcal{K}, t, \sim_{\text{prim}}, g, w)$ *for any* $i \geq n$ *iterations.*

Proof. First, we show that the solution of $\sim_{\text{imax}}$ for $(\mathcal{I}_{Q,\mathcal{T}}, d_Q)$ and $(\mathcal{I}_{\mathcal{K}}, d_a)$ corresponds to the maximal similarity between Q and all concepts D that have a as an instance. This is due to the fact that all concepts D that have a as an instance must be equivalent to generalized concepts of the (possibly infinite) fully expanded $\text{msc}_{\mathcal{K}}(a)$ (see [5]) and that the tree unraveling of $(\mathcal{I}_{\mathcal{K}}, d_a)$ yields exactly the fully expanded $\text{msc}_{\mathcal{K}}(a)$ [12, 14]. By choosing the subsets $S_{\text{CN}} \subseteq \text{CN}(q)$ and $S_{\text{SC}} \subseteq \text{SC}(q)$ for each pair of pointed interpretations $p = (\mathcal{I}_{Q,\mathcal{T}}, d)$ and $q = (\mathcal{I}_{\mathcal{K}}, e)$, the algorithm maximizes the similarity over those generalized concepts, and thus, always computes the maximal similarity between Q and all concepts D that have

a as an instance.

1. Soundness: relaxed-instances computes the similarities between all $d \in \mathcal{I}_{Q,\mathcal{T}}$ and $e \in \mathcal{I}_{\mathcal{K}}$ iteratively. Again, this mapping from old to new maxsim values done in each iteration (line 5–7) is a contraction mapping, and therefore we can apply the Banach fixed-point theorem. This yields that the similarity values computed by relaxed-instances converge to the solutions of $\sim_{\text{imax}}$ and thus for the pair (d_Q, d_a) to the maximal similarity between Q and all concepts D that have a as an instance. Furthermore, all factors used in updating the similarity values are positive, thus the mapping is monotone, and since relaxed-instances starts with similarity value 0 for all pairs of elements, the values for (d_Q, d_a) converges to the solution from below. This means that whenever relaxed-instances finds a value $(d_Q, d_a) > t$, we know that also $(\mathcal{I}_{Q,\mathcal{T}}, d_Q) \sim_{\text{imax}} (\mathcal{I}_{\mathcal{K}}, d_a) > t$ and thus $a \in \text{Relax}_t^{\sim_c}(Q)$. The claim follows.

2. Completeness: Let a be a relaxed instance of Q w.r.t. $\sim_c$, $\mathcal{K}$ and t, i.e. $(\mathcal{I}_{Q,\mathcal{T}}, d_Q) \sim_i (\mathcal{I}_{\mathcal{K}}, d_a) - t = \delta > 0$. The convergence of the similarities computed during relaxed-instances by the Banach fixed-point theorem means that there is an $n \in \mathbb{N}$ such that the error for all iterations $i \geq n$ is less than δ; and thus $a \in$ relaxed-instances$(Q, \mathcal{K}, t, \sim_{\text{prim}}, g, w)$ for all $i \geq n$ iterations. $\square$

Furthermore, the algorithm converges quite fast: For any iteration, the difference between the actual similarity and the computed value reduces by a factor of w. This is again a direct consequence of the Banach fixed-point theorem. This means that, to reduce the error tolerance of the solutions by a constant factor, e.g. one tenth, only a constant number of iterations is needed additionally. However, one cannot compute how many iterations are needed beforehand and cannot be sure if, at any given point, the algorithm already found all relaxed instances, or if some relaxed instances with a maximal similarity very close to the threshold t are still missing.

If applying the algorithm relaxed-instances w.r.t. unfoldable TBoxes $\mathcal{T}$, then the similarities computed in relaxed-instances will however be the exact solutions after exactly k iterations, where $k = \text{rd}(Q) + 1$ is the role-depth of the query concept Q expanded w.r.t. $\mathcal{T}$. In this case, the algorithm can be made deterministic and, since each iteration of relaxed-instances only takes polynomial time in the size of $\mathcal{K}$ and Q, runs in PTIME.

5 Conclusions

We have investigated a new reasoning service that allows relaxed instance query answering for application-specific notions of similarity by the appropriate choice of a CSM—as recently proposed in [5]. The inference has two main degrees of freedom: in the choice of the CSM, and in the degree of relaxation of the concept by the supplied threshold t. Intuitively, different concept similarity measures yield different weights on specific criteria. For example, one could require that small changes inside existential restrictions produce a high level of dissimilarity.

This report extends the inference relaxed instance query answering to the case of general $\mathcal{EL}$-TBoxes. Furthermore, we identified a parameterizable CSM to be employed in this setting. This CSM is derived from a similarity measure for finite (pointed) interpretations, called $\sim_i$. We considered the canonical models of $\mathcal{EL}$-TBoxes (or KBs), which are finite and can be computed in polynomial time. We rephrased formal properties for CSMs for ISMs and developed the parameterizable ISM $\sim_i$ that enjoys many of these properties. Essentially, all properties shown for $\sim_i$ transfer to $\sim_c$. The $\sim_c$ CSMs are, to the best of our knowledge, the first CSMs that take the whole information from general TBoxes into account. Based on $\sim_c$ we gave an computation algorithm for relaxed instances w.r.t. general $\mathcal{EL}$-TBoxes.

There are many options for future work. On the theoretical side it would be interesting to explore how this approach can be extended to expressive DLs. We conjecture that our approach extends to Horn-DLs, since they induce finite canonical models as well. How to generalize our approach and the computation of relaxed instances to DLs that offer all Boolean operators is not obvious.

Regarding the relaxed instances a ranking function of the ascertained individuals would be of interest to return the more interesting relaxed matches first to such a query. Here, a natural idea is, of course, to rank the individuals (or their msc resp.) according to the similarity to the query concept.

On the practical side there is plenty of room for optimizations. For instance, the use of a concept that states necessary conditions in combination with the query concept can considerably reduce the number of individuals to be checked in practice. Furthermore, while the complexity of each iteration in the general case is polynomial, the need to check every subset and every pairing is certainly inefficient. Methods to reduce the subsets and pairings that need to be considered are expedient to make this work in practice.

References

[1] M. A. Alvarez and C. Yan. A graph-based semantic similarity measure for the gene ontology. *J. Bioinformatics and Computational Biology*, 9(6):681–695, 2011.

[2] F. Baader, R. Küsters, and R. Molitor. Computing least common subsumers in description logics with existential restrictions. In T. Dean, editor, *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI'99)*, pages 96–101. Morgan Kaufmann, 1999.

[3] S. Banach. Sur les oprations dans les ensembles abstraits et leur application aux quations intgrales. *Fundamenta Mathematicae*, 3(1):133–181, 1922.

[4] C. d'Amato, S. Staab, and N. Fanizzi. On the influence of description logics ontologies on conceptual similarity. In A. Gangemi and J. Euzenat, editors, *Proceedings of Knowledge Engineering: Practice and Patterns, 16th Int. Conf. (EKAW 2008)*, volume 5268 of *LNCS*, pages 48–63. Springer, 2008.

[5] A. Ecke, R. Peñaloza, and A.-Y. Turhan. Towards instance query answering for concepts relaxed by similarity measures. In *Workshop on Weighted Logics for AI (in conjunction with IJCAI'13)*, Beijing, China, 2013.

[6] T. Gene Ontology Consortium. Gene Ontology: Tool for the unification of biology. *Nature Genetics*, 25:25–29, 2000.

[7] M. R. Henzinger, T. A. Henzinger, and P. W. Kopke. Computing simulations on finite and infinite graphs. In *IEEE Symposium on Foundations of Computer Science*, pages 453–462, 1995.

[8] R. Küsters and R. Molitor. Approximating most specific concepts in description logics with existential restrictions. *AI Communications*, 15(1):47–59, 2002.

[9] K. Lehmann and A.-Y. Turhan. A framework for semantic-based similarity measures for $\mathcal{ELH}$-concepts. In L. F. del Cerro, A. Herzig, and J. Mengin, editors, *Proceedings of the 13th European Conference on Logics in Artificial Intelligence*, Lecture Notes in Artificial Intelligence, pages 307–319. Springer Verlag, 2012.

[10] C. Lutz and F. Wolter. Deciding inseparability and conservative extensions in the description logic $\mathcal{EL}$. *Journal of Symbolic Computation*, 45(2):194–228, 2010.

[11] M. Mistry and P. Pavlidis. Gene ontology term overlap as a measure of gene functional similarity. *BMC Bioinformatics*, 9, 2008.

[12] R. Peñaloza and A.-Y. Turhan. A practical approach for computing generalization inferences in $\mathcal{EL}$. In M. Grobelnik and E. Simperl, editors, *Proceedings of the 8th European Semantic Web Conference (ESWC'11)*, Lecture Notes in Computer Science. Springer-Verlag, 2011.

[13] A. Schlicker, F. S. Domingues, J. Rahnenführer, and T. Lengauer. A new measure for functional similarity of gene products based on gene ontology. *BMC Bioinformatics*, 7:302, 2006.

[14] B. Zarrieß and A.-Y. Turhan. Most Specific Generalizations w.r.t. General $\mathcal{EL}$-TBoxes. In F. Rossi, editor, *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, 2013.